

| **1. Resumen avance Proyecto APT** |
| --- |

| Resumen de avance proyecto APT | El proyecto "BRP", para la gestión integral de proyectos contratistas, ha alcanzado un 56% de avance. Se han completado exitosamente las etapas de análisis de requerimientos, el diseño de una arquitectura de software de tres capas, el desarrollo de la base de datos relacional en PostgreSQL/Supabase, y la construcción de los módulos fundamentales del backend (Node.js) y frontend (React.js). Actualmente, el equipo se enfoca en la integración final de ambos componentes, la ejecución de pruebas funcionales exhaustivas y la elaboración de la documentación técnica. El desarrollo ha seguido los plazos planificados en Trello, gracias a una coordinación efectiva y al uso de metodologías ágiles. |
| --- | --- |
| Objetivos | Los objetivos fundamentales establecidos al inicio del proyecto se han mantenido firmes y sin modificaciones. La visión original no ha cambiado; en su lugar, la comprensión sobre cómo alcanzarla se ha detallado a medida que el proyecto avanza. Las funcionalidades desarrolladas hasta ahora abordan directamente los objetivos iniciales, como el registro de clientes, la creación y control de proyectos, y la gestión de recursos. |
| Metodología | La metodología de trabajo se ha mantenido basada en Scrum con un enfoque de desarrollo iterativo. Aunque la estructura de Sprints y los roles definidos no han sufrido modificaciones, se realizó un ajuste en el alcance técnico del proyecto. Específicamente, se incorporó un nuevo procedimiento para reforzar la seguridad de la aplicación mediante la implementación de Cloudflare, una tarea que no estaba contemplada en la planificación original pero que fue integrada eficientemente dentro de los Sprints existentes sin alterar el plazo de entrega final. |
| Evidencias de avance | Las evidencias que demuestran el desarrollo y la calidad del proyecto son las siguientes:   * **Repositorio en GitHub**: Contiene todos los commits y el código fuente del proyecto, reflejando el trabajo de programación continuo. * **Captura del tablero de Trello**: Muestra la planificación, asignación y el estado de avance de todas las tareas gestionadas por el equipo. * **Scripts SQL y Diagrama MER**: Justifican el diseño y la implementación de la base de datos relacional, que es el núcleo del sistema. * **Documentación técnica (Requerimientos, Arquitectura, Bitácora de pruebas)**: Estos documentos demuestran que el desarrollo se basa en un análisis sólido y sigue las buenas prácticas de la disciplina, asegurando la calidad y mantenibilidad del software. |
| **2. Monitoreo del Plan de Trabajo** |
|  |

| **3. Ajustes a partir del monitoreo** |
| --- |

| Factores que han facilitado y/o dificultado el desarrollo de mi plan de trabajo  **Facilitadores:**   * **Metodología Ágil (Scrum)**: Ha permitido una gran flexibilidad para adaptar el plan de trabajo, como la integración de nuevas tareas de seguridad sin afectar el cronograma. * **Herramientas de Gestión (Trello y GitHub)**: El uso de Trello facilitó una clara organización y seguimiento de las tareas, mientras que GitHub fue clave para la colaboración y el control de versiones del código. * **Stack Tecnológico Moderno**: El uso de React.js con Vite proporcionó un entorno de desarrollo rápido y eficiente, agilizando la construcción del frontend. * **Buena Coordinación de Equipo**: Una comunicación constante y una clara definición de roles han permitido un flujo de trabajo coordinado y sin mayores contratiempos.   **Dificultades:**   * **Necesidad de mayor detalle en la planificación inicial**: Al inicio, el listado de tareas era muy general. Esto se abordó realizando una revisión exhaustiva en equipo y desglosando las metas en tareas más específicas y manejables en Trello, lo que resultó en una planificación actualizada mucho más precisa. * **Requerimiento de seguridad no contemplado**: Se identificó la necesidad de reforzar la seguridad del sistema, lo cual no estaba en el plan original. Esto se soluciona integrando nuevas tareas para la implementación de Cloudflare dentro de los Sprints actuales, aprovechando la flexibilidad de Scrum. |
| --- |

| Actividades ajustadas o eliminadas:  El principal ajuste al plan de trabajo no fue eliminar actividades, sino **ampliar y detallar el listado de tareas inicial**. Tras una reevaluación del equipo, se concluyó que el plan original era demasiado general. Por ello, se desglosaron los objetivos en tareas más granulares y específicas para asegurar que todos los requerimientos fueran abordados de manera precisa.  Adicionalmente, se **añadieron nuevas actividades** que no estaban contempladas originalmente:   * **Implementación de seguridad con Cloudflare**: Se decidió añadir esta capa de seguridad para robustecer la aplicación de cara a su despliegue. Esta actividad fue integrada en la planificación existente sin alterar la fecha final de entrega. * **Rediseño general y modo claro/oscuro**: Se decidió realizar una mejora visual sobre el diseño base existente para potenciar la experiencia de usuario.   No se eliminó ninguna actividad, ya que todas las planificadas originalmente seguían siendo esenciales para cumplir con los objetivos del proyecto. |
| --- |

| Actividades que no has iniciado o están retrasadas  Actualmente no hay actividades con retraso. Las tareas pendientes de iniciar, como la "Integración de Seguridad en Back-End", el "Desarrollo de Dashboards de Documentos" o la "Documentación de Seguridad y Roles", están programadas para comenzar en las próximas semanas, de acuerdo con la planificación actualizada en Trello.  La estrategia para asegurar que estas tareas se completen a tiempo consiste en mantener la disciplina de la metodología Scrum, realizando seguimientos diarios (daily stand-ups) para comunicar avances y posibles bloqueos, y respetando los plazos definidos para cada Sprint. El equipo continuará utilizando Trello para visualizar el progreso y garantizar que el proyecto avance según lo planeado hacia la entrega final. |
| --- |